Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 24 (2013) 261 – 267

Procedia
Computer Science

17th Asia Pacific Symposium on Intelligent and Evolutionary Systems, IES2013

Applying Non-negative Matrix Factorization to Classify Superimposed Handwritten Digits

Somnuk Phon-Amnuaisuk[a,*]

[a]*Faculty of Business and Computing, Brunei Institute of Technology, Brunei*

Abstract

Computation models that rely on a decision boundary approach have a severe limitation in recognizing the same objects with different appearances. Variations in the appearance of an object often create variations in their discriminative features as well. Hence, the derived feature data points might be shifted into the wrong decision regions. In this paper, we present an application of a dictionary-based *Non-negative Matrix Factorization (NMF)* for classifying handwritten digits under superimposition. The superimposed digits are encoded as the input V, which is factorized into two matrices, $V \approx WH$ where W is the basis vector representing the prototypes of handwritten digits and H is the encoding coefficients representing the digit classification output sequence. We show that a dictionary-based NMF could classify an input sequence of single handwritten digits as well as superimposed digits. This is because NMF searches for an optimal linear aggregation of the components in the dictionary $V \approx WH$. This unique ability of NMF allows it to classify superimposed digits. The experimental results show that the proposed approach can recognize superimposed digits with good accuracy.

Selection and peer-review under responsibility of the Program Committee of IES2013

Keywords: Recognition of superimposed handwritten digits; Non-negative matrix factorization

1. Background

It is widely accepted that human visual process starts off from edge detection, then component segmentation before perceptual interpretation[1]. Our visual systems possess the ability to resolve ambiguities introduced from variations such as lighting, orientation, scale, superimposition and occlusion. Although there are some guiding principles in perceptual interpretation such as Gestalt principles, to our best knowledge, there is no known computation technique that could mimic perceptual behaviors such as the ability to filter out only the interested region in superimposition cases. This is still an open research issue.

Common computational approaches employed in image recognition tasks are often based on decision boundary of classification models and probabilistic models. These approaches have a severe limitation in recognizing variants of the same object since the decision boundary is constructed from training examples. An object with superimposition or occlusion will appear as a different object to the model learned from examples.This limitation is also common in

* Corresponding author
E-mail address: somnuk.phonamnuaisuk@itb.edu.bn

Selection and peer-review under responsibility of the Program Committee of IES2013
doi:10.1016/j.procs.2013.10.049

dictionary-based template matching approaches. In this work, however, we investigate the application of a *dictionary-based NMF* to tackle the superimposed digits issue. The dictionary-based NMF recognizes objects by looking up for a match from the aggregations of components in the dictionary. The approach is chosen since it offers a unique ability to infer linear aggregation of dictionary components. This means the approach should be able to handle superimposition cases. This work attempts to investigate this issue.

In our application, the matrix W is initialized as a dictionary of handwritten digits. Hence, a successful factorization of a sequence of input handwritten digits, V, yields a sequence of digit classification output H. With this approach, a sequence of digits or superimposed digits can still be classified correctly.

The rest of the paper is organised as follows: section 2 presents the related works, discusses our motivations and explains the problem formulation; section 3 presents the important concepts of the proposed approach; section 4 presents the experimental design and discusses the experimental results; and finally, section 5 concludes this research work.

2. Related Work

Recognizing handwritten digits is a sub-area of handwritten recognition research. A complete handwriting recognition system processes either online or offline handwritten input by segmenting handwritten input into characters, recognizing each character and producing the results in a suitable format. Various application domain of handwritten symbols have been studied, for examples: reading postal address[2]; reading vehicle license plates[3]; reading music notation[4]; and reading mathematical symbols[5].

Classification models and probabilistic graphical models are the common approaches employed in handwritten digit recognition. The decision regions of these models are constructed from many training examples. As long as the constructed models do not over-fit the data, the more the examples, the more robust the model is. Bottou et al.[6] shows that *convolution neural nets* can handle large training data better than many other techniques e.g., linear classifier, polynomail classifier and support vector machine. To date, the convolution neural net has the smallest error rate in classifying MNIST handwritten digit data set, that is less than 0.5 %.

All classification techniques that rely on feature matching between an input query and stored patterns are, in principle, a dictionary lookup approach. The performance of the systems in this category depends on the completeness of the dictionary. How could the dictionary be extended to handle variations introduced to the system? Instead of creating more classes to handle different varients. A more effective approach would be to infer about each different model and results from the interactions of different models, since the number of models are much lesser than the number of possible variations. In the digits superimposition case, NMF provides the means for explicit reasoning about each digit and superimposed digits.

NMF has been investigated by[7,8,9]. NMF offers a part-based interpretation of local components which are in contrast with other holistic approaches where local features are not explicitly encoded by default. The idea of a part-based aggregation in NMF is useful and NMF has been applied to many domains such as facial recognition[8], image segmentation[10], document categorization, clustering[9], and polyphonic transcription[11,12]. In the recent work by[13], the localized NMF is employed to recognize occluded images.

In essence, NMF factorizes an input matrix V into a linear combination of two components W and H, i.e., $V^{m\times n} \approx W^{m\times r}H^{r\times n}$. There are many plausible combinations of W and H for any given V, even though the non-negativity constraint is applied. Other domain dependent constraints may be added to guide the factorization process.

3. A Dictionary-based Non-negative Matrix Factorization

Superimpositions are common in visual analysis. Superimposed symbols are common in graphic designs but the issue has been somewhat neglected by researchers. To the best of our knowledge, there is no standard data set for the superimposition problem. Here, we performed experiments on MNIST handwritten digits dataset. This data set is a collection of handwritten digits from 500 subjects. It comprises 60,000 training examples and 10,000 testing examples, roughly 6,000 training examples and 1,000 testing examples for each digit. Each example is represented in 28×28 pixels and each pixel has 256 intensity levels (more details of MNIST dataset can be accessed via http://yann.lecun.com/exdb/mnist/).

(a) A subset of digits from the MNIST database.

(b) The third row shows superimposition of digits from the first and the second row.

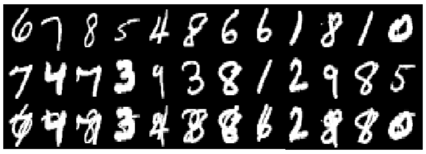

Fig. 1. (a) Examples from the MNIST database; and (b) Superimposition of two digits

Figure 1 shows the samples of digits from the MNIST database and the samples of superimposed digits. We construct the superimposed digits from MNIST data set by compositing two digits together. The intensity of the pixels in the superimposed image is the maximum intensity from the two digits.

3.1. Using the Dictionary-based NMF as a Classifier

Let $V^{m\times n}$ be a matrix where $\mathbf{v}_n$ is a column vector representing the features extracted from a single handwritten digit. Here, the representation of each digit in NMF is obtained by mapping each pixel p_{ij} of a digit $\mathcal{D}^{r\times c}$, where $i = 1..r$; $j = 1..c$, to v_k, s.t. $k = j + (i - 1)c$. This forms a column vector $\mathbf{v} \in [0, 255]^{784\times 1}$. In other words, this is a flattening of each row in the matrix $\mathcal{D}$ to form the vector $\mathbf{v}$. Due to the non-negativity constraint of NMF, a further dimension reduction using techniques such as the Principle Component Analysis (PCA) is not possible since PCA produces negative values.

The non-negative constraint of NMF gives an interpretation of a linear aggregation of the basis column vectors W and the coefficients of the linear aggregation H that could reconstruct the original data V. However, there could be many possible W and H that could satisfy $V \approx WH$. In most applications, a constraint on the number of basis vectors $\mathbf{w}_r$ is imposed. In our approach, each basis vector $\mathbf{w}_r$ is the prototype of a handwritten digit. We impose constraints on both the content and the size of W.

Since each handwritten digit class can have many variations (it is writer-dependent), we have decided to cluster the training examples based on the Euclidean distance into k clusters (for each digit class) and take the center of each cluster to be the prototype digit of that cluster. Let $W^{m\times r}$ be a matrix where $\mathbf{w}_r$ is a column vector representing the features of a digit prototype. The matrix W is the dictionary matrix.

For each test run, we randomly pick 2000 digits from the test data set. This forms a matrix $V^{784\times 2000}$. The corresponding class labels $L \in \{0, 1, ..., 9\}$ of all the digits in V are recorded to provide a ground truth for analyzing the classification results. Each entry in the ground truth matrix $G \in \{0, 1\}^{r\times n}$ indicates whether the corresponding entry in the dictionary is True (1) or False (0). The entry is set to 1 if the selected digit is from that entry, otherwise it is set to 0. The size of the matrix G is the same as the size of the matrix H where each row corresponds to an entry in the dictionary W and each column corresponds to each input digit $\mathbf{v}$.

Given a matrix V and the dictionary W, we could compute H by minimizing the cost function below[8]:

$$D_F(V||WH) = ||V - WH||_F^2 = \sum_{mn}(V_{mn} - (WH)_{mn})^2 \tag{1}$$

where $||.||_F$ denotes the Frobenius norm and V_{mn} stands for the mn^{th} entry of V. In this work, we implement the multiplicative update rule modified from[8]. The matrix W is fixed and the matrix H is iteratively updated as follows:

$$H_{rn} \leftarrow H_{rn} \frac{\sum_m W_{mr} V_{mn}/(WH)_{mn}}{\sum_{m'} W_{m'r}} \tag{2}$$

The multiplicative update rule above modifies each entry rn^{th} in the matrix H based on the contribution of $\mathbf{w}_r$ and the n^{th} column vector from $V_{mn}/(WH)_{mn}$ (i.e., $\sum_m W_{mr} V_{mn}/(WH)_{mn}$). The term $\sum_{m'} W_{m'r}$ acts as a normalization factor. If WH successfully approximates V, then W and H are at the stationary point which is the local-optimal solution.

We hypothesize that our computational model can handle the superimposition case. Employing a digit-model dictionary in NMF makes NMF a kind of generative model. The matrix V could be thought of as being constructed from a linear aggregation of a dictionary W and the activation coefficients H. In other words, the H matrix reveal the class of the input matrix V. In our setup, the matrix $V^{m \times n}$ represents m features of n digit sequence, while the dictionary $W^{m \times r}$ represents m features of r entries in the dictionary, and $H^{r \times n}$ represents the classification output for a sequence of n digits.

Figure 2 summarizes the ideas behind our approach: (i) The dictionary W is constructed using k cluster centers of each digit from the MNIST training data set; (ii) A test sequence V is constructed by randomly picking 2000 digits from the test data set (for the superimposition of 2 digits, 4000 digits will be randomly selected). The class labels are recorded for comparison to the classification results. The system factorizes the input V using the constraint from W to obtain the classification H. Each column of $\mathbf{h}_n$ shows the contribution of the entry r of $\mathbf{w}_r$ towards the input $\mathbf{v}_n$. For a single digit, the classification is interpreted from $argmax\ \ \mathbf{h}_n$. For the superimposition of two digits, the first two highest entries are interpreted as the classified digits.

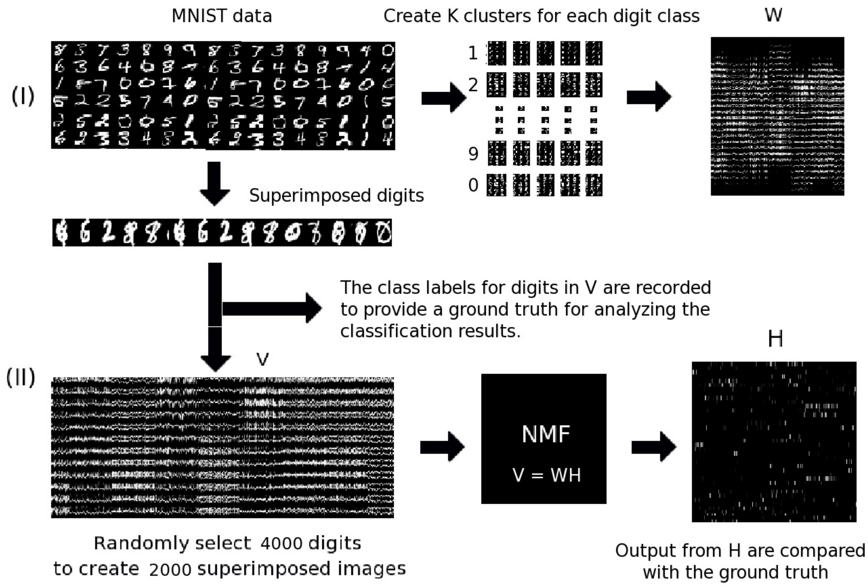

Fig. 2. Basic concepts of Dictionary-based NMF (see text for more explanation)

3.2. Experimental Design

One drawback of the NMF approach is the computation cost from matrix operations. It will be too expensive to include all available training examples in the dictionary-based NMF approach. The strings of pixels expanded from a 28×28 pixels image require a dictionary matrix having 784×60000 entries. We have decided to cluster handwritten digit data using the k-means clustering technique, with the following number of clusters $k = \{10, 30, 50, 70, 90\}$. The dictionary is built using only the digits obtained from the cluster center. Hence, for $k = 90$, there will be 900 digits in the dictionary, i.e., 784×900 entries.

There were two main parts in this experiment. In the first part, we show that a dictionary-based NMF can perform a dictionary lookup task and its non-negative aggregation enables a complex lookup involving superimposed digits. To provide a baseline for comparison with further experiments, the test data from the first part is drawn from the dictionary. As a consequence, we would expect to see a stable performance of the models in terms of classification accuracy over different dictionary sizes. In the second part, we investigate an inexact dictionary lookup for single digits and superimposed digits. Here, the dictionary is built from the training set and the test data from the test data set. The inexact dictionary lookup shows how well a dictionary-based representation perform as a generalized model.

4. Results and Discussion

In our experiments, the input V is a sequence of randomly picked digits from the test pool (i.e., the cluster centers for the first part and the digits from the test set for the second part of the experiment). The accuracy is computed by counting the number of the correctly classified digits in the matrix H with reference to the ground truth matrix G. The values reported for all experiments are averaged over 10 runs.

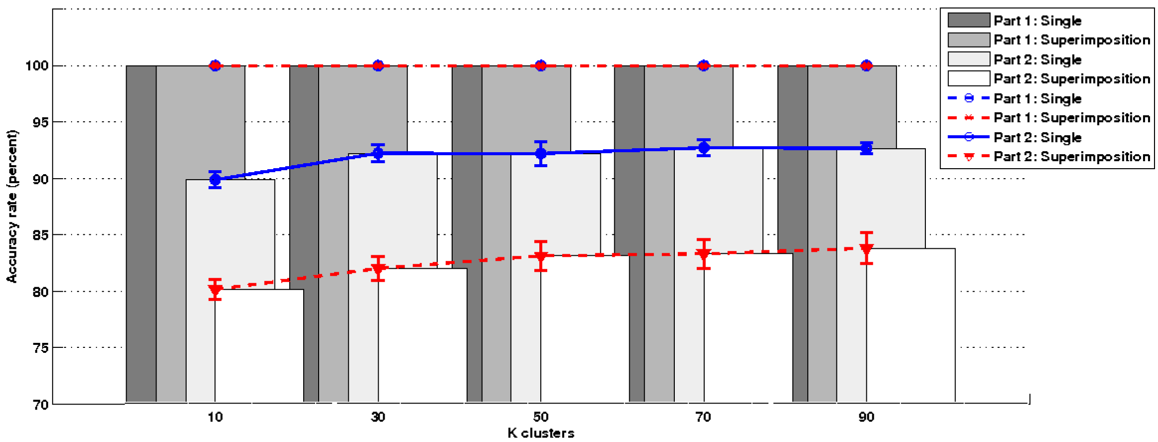

Fig. 3. The results from *Part 1* show the baselines of the models. They have 100% accuracy for all k values. The generalization accuracy of the models are shown in *Part 2* results (take note that the scale on the Y-axis starts from 70). Error bars show the standard deviation values from the mean values.

We measure the performance of the system using the accuracy of the models which are computed from the number of correct classifications. Figure 3 shows that the output from the first part of the experiment are stable across different values of k. This is expected since increasing the size of the dictionary should not jeopardize the accuracy. The results confirm that the dictionary-based NMF can perform the lookup task. This gives the performance baselines under the current implementation for a single digit case and a superimposed digits case.

For the second part of the experiment, the accuracy of the models in generalizing single digits are around 90-93% and around 80-84% for superimposed digits. There is a clear increment in accuracy rates when k increases from 10 to 30. However, there seems to be little improvement in the classification accuracy for k higher than 50. Although the accuracy from the baseline cases implies that the system can perform well if the dictionary is complete, it is impractical to keep on increasing the number of clusters since the computing cost of the NMF outweighs the gain in accuracy improvement.

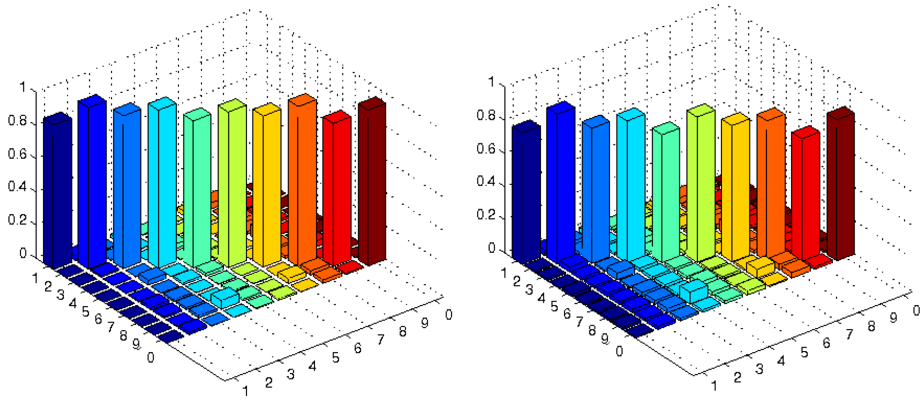

Fig. 4. The average accuracy of the models broken down in terms of each digit; the left pane shows the single digit case and the right pane shows the superimposed digits case. The X-axis shows the actual class while the Y-axis shows the classification output and the Z-axis shows the accuracy (×100). The axes are in the right-handed system.

Figure 4 shows the output from the second part of the experiment. We show the average accuracy of the models broken down for each digit. The output from single digits has a higher accuracy than those from superimposed digits. The overall performance of the dictionary-based NMF approach is stable, as is evident from the low standard deviations across the different values of clusters.

It is interesting to note that both output share the same patterns, i.e., the even digits have a slightly higher accuracy than the odd digits. Apparently NMF finds it easier to classify the even digits. We believe this results from the ambiguity of the shapes of odd digits, for example, the handwritten digit 9 could be quite similar (for NMF) to the handwritten digits of 4, 5, 7 and probably 1; while the handwritten digit of 2 appears to be less ambiguous.

5. Conclusion and Future Work

In this paper, we have shown that a dictionary-based NMF can perform a lookup task. Its non-negative factoring offers a linear aggregation of the components in the dictionary. This allows the dictionary-based NMF to look up digits superimposed on top of each other. This type of superimposition can be observed in a graphic design domain where the texts are often superimposed on other objects (which could be texts with different fonts or images). This problem is very hard, if not impossible, for standard pattern recognition techniques that learn to classify features of different classes by learning decision boundaries of classes.

The accuracy of the single digit case is poorer than the current state of the art techniques, such as convolution neural network. However, this work offers a novel usage of NMF to separate superimposed objects by relying on the non-negative linear aggregation of NMF. We hope this could spark interests in solving superimposition and occlusion problems using component aggregation techniques. In future work, we hope to improve on the computing efficiency and accuracy of classification through a more expressive representation of features.

Acknowledgements

We wish to thank Brunei Institute of Technology for their partial financial support given to this research. We would also like to thank anonymous reviewers for their constructive comments and suggestions.

References

1. Steven, S.H. and Jolij, J. and Lamme, V.A.F., The cortical processing dynamics of edge detection and scene segmentatation, In H. Ögmen and B.G. Breitmeyer, (Eds.), *The First Half Second: The Microgenesis and Temporal Dynamics of Unconscious and Conscious Visual processes*. MIT Press; 2006. p. 73-87.
2. Kimura, F. and Miyake, Y. and Shridhar, M., Handwritten ZIP code recognition using lexicon free word recognition algorithm In *Proceedings of the 3rd International Conference on Document Analysis and Recognition*; 1995. p. 906-910.
3. Comelli, P. and Ferragina, P. and Granieri, M.N. and Stabile, F., Optical recognition for motor vehicle license plates. *IEEE Transactions on Vehicular Technology* 1995; **44** (4): 790-799.
4. Lee, K.C. and Phon-Amnuaisuk, S. and Ting, C.Y., Handwritten music notation recognition using HMM: A non-gestural approach, In *Proceedings of the International Conference on Information Retrieval & Knowledge Management, (CAMP 2010)*; 2010. p. 255-259.
5. Matsakis, N.E., Recognition of handwritten mathematical expressions Master of Engineering in Electrical Engineering and Computer Science. Massachusetts Institute of Technology; 1999.
6. Bottou, L. and Cortes, C. and Denker, J.S. and Drucker, H. and Guyon, I. and Jackel, L.D. and LeCun, Y. and Müller, U.A. and Säckinger, E. and Simard, P. and Vapnik, V., Comparison of classifier methods: A case study in handwritten digit recognition. In *Proceedings of the 12th IAPR International. Conference on Pattern Recognition*, Vol. 2 - Conference B: Computer Vision & Image Processing; 1994.
7. Paartero, P. and Tapper, U., Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Envirometrics* 1994; **5**(2):111-126.
8. Lee, D.D. and Seung, H.S., Learning the parts of objects by non-negative matrix factorization. *Nature* 1999; **401**: 788-791.
9. He, Z.S. and Xie, S.G. and Zdunek, R. and Zhou, G.X. and Cichocki, A., Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering *IEEE Transactions on Neural Networks* 2011; **22**(12) Part: 1: 2117-2131.
10. Sandler, R. and Lindenbaum, M., Nonnegative matrix factorization with earth mover's distance metric for image analysis *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2011; **33**(8): 1590-1602.
11. Smaragdis, P. and Brown, J.C., Non-negative matric factorization for polyphonic music transcription In *Proceedings of IEEE workshop Applications of Signal Processing to Audio and Acoustics.* New Paltz, NY; 2003. p. 177-180.
12. Phon-Amnuaisuk, S., Transcribing Bach chorales: Limitations and potentials of non-negative matrix factorisation, *EURASIP J. Audio, Speech and Music Processing* 2012; **(2012)** 11.
13. Bajla, I. and Soukup, D. and Stolc, S., Occluded image object recognition using localized nonnegative matrix factorization methods, In: T.P. Cao (Ed.), *Object Recognition*. Ch. 6; 2011. p. 83-105.